

BAB II

LANDASAN TEORI

2.1 Sistem

Definisi Sistem Menurut Jogianto (2005 : 2) mengemukakan bahwa sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

Suatu sistem pada dasarnya adalah sekolompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu.

Dari defenisi ini dapat dirinci lebih lanjut pengertian sistem secara umum, yaitu :

1. Setiap sistem terdiri dari unsur-unsur.
2. Unsur-unsur tersebut merupakan bagian terpadu sistem yang bersangkutan.
3. Unsur sistem tersebut bekerja sama untuk mencapai tujuan sistem.
4. Suatu sistem merupakan bagian dari sistem lain yang lebih besar.

2.2 Informasi

Menurut Jogiyanto HM. dalam bukunya yang berjudul Analisis dan Desain Sistem (2005 : 8), menyebutkan bahwa: "Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan berarti bagi yang menerimanya". Menurut Tata Sutabri dalam bukunya yang berjudul Analisa Sistem Informasi (2003 : 18), menyebutkan bahwa : "Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan". Berdasarkan definisi di atas penulis dapat menarik kesimpulan bahwa informasi adalah data yang sudah diolah menjadi sesuatu yang bermanfaat untuk pemakainya.

Menurut Tata Sutabri dalam bukunya yang berjudul Analisa Sistem Informasi (2003 : 30), menyebutkan bahwa kualitas informasi adalah sebagai berikut:

- a. Akurat (*accurate*), informasi harus bebas dari kesalahan dan tidak menyesatkan.
Informasi ini harus mencerminkan keadaan yang sebenarnya.

- b. Tepat Waktu (*TimeLines*), informasi yang sampai pada si penerima tidak boleh terlambat, harus tersedia pada saat informasi tersebut diperlukan.
- c. Relevan (*relevance*), informasi yang diberikan harus sesuai dengan yang dibutuhkan dan mempunyai manfaat untuk pemakainya.

Berdasarkan penjelasan diatas maka penulis dapat menarik kesimpulan bahwa suatu informasi dapat dikatakan berkualitas jika memenuhi kualitas informasi yaitu akurat, tepat waktu dan relevan.

2.3 Sistem Informasi

Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses algoritmik, data, dan teknologi. Dalam pengertian ini, istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi (TIK), tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis.

Ada yang membuat perbedaan yang jelas antara sistem informasi, dan komputer sistem TIK, dan proses bisnis. Sistem informasi yang berbeda dari teknologi informasi dalam sistem informasi biasanya terlihat seperti memiliki komponen TIK. Hal ini terutama berkaitan dengan tujuan pemanfaatan teknologi informasi. Sistem informasi juga berbeda dari proses bisnis. Sistem informasi membantu untuk mengontrol kinerja proses bisnis.

Sistem suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

2.4 Pengarsipan

Menurut Gie (1998) mengungkapkan bahwa arsip adalah kumpulan warkat yang disimpan secara teratur berencana karena mempunyai suatu kegunaan agar setiap kali diperlukan dapat cepat ditemukan kembali.

Menurut Hasugian (2012) mengungkapkan bahwa dalam pemahaman sederhana dapat dinyatakan bahwa arsip adalah merupakan salah satu produk kantor (office work). Artinya, kearsipan merupakan salah satu jenis pekerjaan kantor atau pekerjaan tatausaha, yang banyak

dilakukan oleh badan-badan pemerintah, maupun badan swasta. Kearsipan menyangkut pekerjaan yang berhubungan dengan penyimpanan warkat atau surat-surat, dan dokumen-dokumen kantor lainnya.

2.5 Dokumen

Dokumen adalah menurut bahasa Inggris berasal dari kata “*document*” yang mempunyai arti suatu yang tertulis atau yang tercetak dan segala benda yang memiliki berbagai keterangan dipilih untuk di disusun, di kumpulkan, di sediakan ataupun untuk disebarluaskan.

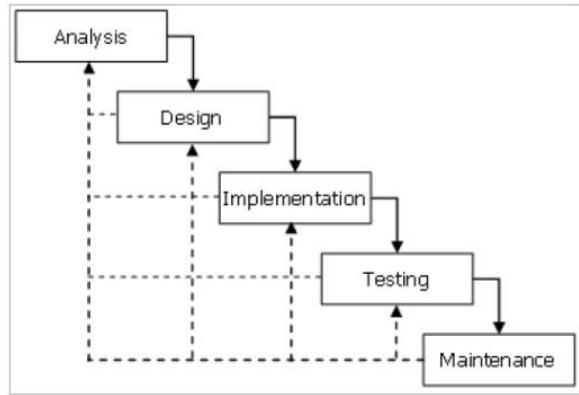
2.6 Metode Pengembangan

Metode pengembangan system adalah sekumpulan langkah, prosedur, konsep maupun aturan dalam mengembangkan sebuah system. Tahap ini menjelaskan alur dan tahapan yang dilakukan dalam pengembangan sistem. Melalui tahapan ini, diharapkan sistem yang dikembangkan dapat terselesaikan dan sesuai dengan sasaran yang ditetapkan. Penerapan metode/model dalam pengembangan sistem juga dapat digunakan sebagai kontrol kerja.

Metode/model SDLC (*System Development Life Cycle*) dengan model proses air terjun (*waterfall*) atau lebih dikenal dengan istilah siklus kehidupan klasik. Air terjun, ciri khas dari air terjun adalah aliran searah dari atas ke bawah secara teratur. Begitu juga dengan model ini, setiap fase/tahap dalam SDLC waterfall harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase/tahap berikutnya.

Menurut Bassil (2011), Model Waterfall SDLC adalah proses pengembangan perangkat lunak yang berurutan (sequential) dimana prosesnya dari atas ke bawah (seperti air terjun) melalui tahapan-tahapan yang harus dijalankan untuk keberhasilan pembuatan perangkat lunak.

Metode tahapan pengembangan sistem SDLC (*System Development Life Cycle*) model *Waterfall* ada pada Gambar 2.1 berikut :



Gambar 2.1 Sistem SDLC (*System Development Life Cycle*) model *Waterfall*

(Sumber : Youssef Bassil, 2011)

2.6.1 Fase Analisa (*Analisa Phase*)

Fase analis sering disebut juga sebagai spesifikasi kebutuhan perangkat lunak (*Software Requirements Specification* atau *SRS*), yaitu deskripsi lengkap dan komprehensif tentang perilaku perangkat lunak yang dikembangkan.

2.6.2 Fase Desain (*Design Phase*)

Fase desain adalah proses perencanaan dan pemecahan masalah (*problem solving*) untuk sebuah solusi perangkat lunak.

2.6.3 Fase Implementasi (*Implementation Phase*)

Fase implementasi mengacu pada realisasi kebutuhan bisnis dan spesifikasi desain ke dalam bentuk program nyata, *database*, *website*, atau komponen perangkat lunak melalui pemrograman dan penempatan (*deployment*).

2.6.4 Fase Pengujian (*Testing Phase*)

Fase pengujian juga dikenal sebagai verifikasi dan validasi, yaitu sebuah proses untuk memeriksa bahwa solusi sebuah perangkat lunak memenuhi persyaratan dan spesifikasi dan itu menyelesaikan tujuan yang telah ditetapkan.

2.6.5 Fase Perawatan (*Maintenance Phase*)

Fase perawatan adalah proses memodifikasi solusi perangkat lunak setelah dibuat dan diterapkan untuk memperbaiki output, memperbaiki *error* dan meningkatkan kinerja dan kualitas.

2.7 *Unified Modeling Language (UML)*

Unified Modeling Language (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Rosa & Shalahudin, 2014).

2.8 *Use Case Diagram*

Menurut Rosa dan Shalahuddin (2014) mengungkapkan *Use Case Diagram* Pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Use Case diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. *Use Case* diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Simbol-simbol yang ada pada *Use Case* diagram dapat dilihat pada tabel 2.2 berikut :

Tabel 2.1 Simbol *Use Case Diagram*

No	Simbol	Keterangan
1.		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor.
2.		Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, walaupun simbol aktor adalah orang namun aktor belum tentu merupakan orang. Biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.

Tabel 2.1 Simbol *Usecase Diagram* (lanjutan)

3.	<i>Association</i>	Komunikasi antara aktor dan <i>Use Case</i> yang berpartisipasi pada usecase atau <i>Use Case</i> memiliki interaksi dengan aktor.
4.	<i>Ekstensi/Extend</i> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>Use Case</i> tambahan itu.
5.	<i>Generalization</i> 	Hubungan generalisasi dan spesialisasi antara dua buah <i>Use Case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
6.	<i>Include</i> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan memerlukan <i>Use Case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>Use Case</i> ini.

(Sumber : Rosa dan Shalahudhin, 2014)

2.9 *Use Case Skenario*

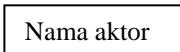
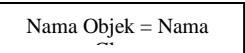
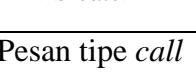
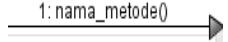
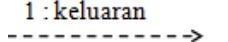
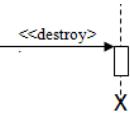
Menurut Rosa dan Shalahuddin(2014) *Use case scenario* sebuah dokumentasi terhadap kebutuhan fungsional dari sebuah sistem. Model *use case* (*use case scenario*) menggambarkan bagaimana sistem bekerja secara keseluruhan dan pada skenario tertentu sehingga jika sebuah *object* merupakan sesuatu yang menyediakan layanan maka harus dapat dibuktikan bahwa semua layanan tersebut ada dalam suatu *use case*.

2.10 *Sequence Diagram*

Menurut Rosa dan Shalahudin(2014) *diagram sequence* menggambarkan kelakuan objek pada *Use Case* dengan mendeskripsikan waktu hidup objek atau message yang dikirimkan dan diterima antar objek.Oleh karena itu untuk menggambarkan diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *usecase* berserta metode-metode yang dimiliki kelas.

Simbol –simbol yang ada pada *sequence diagram* dapat dilihat pada tabel 2.2 berikut ini :

Tabel 2.2 Simbol Sequence Diagram

Simbol	Keterangan
Aktor  atau  Tanpa waktu aktif	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawali <i>frase</i> nama aktor.
Garis hidup 	Menyatakan kehidupan suatu objek
Objek 	Menyatukan objek yang berinteraksi pesan
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirim data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan sesuatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu.
Pesan Tipe <i>Destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

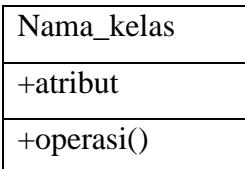
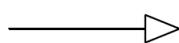
(Sumber: Rosa dan Shalahudin, 2014)

2.11 Class Diagram

Menurut Rosa dan Shalahudin (2014) *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut attribut dan metode atau operasi.

Attribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Simbol – simbol yang ada pada *class diagram* dapat dilihat pada Tabel 2.3 berikut ini :

Tabel 2.3 Simbol *Class Diagram*

Simbol	Keterangan
Kelas/Class 	Kelas pada struktur system
Antar muka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>association</i> 	Relasi antar kelas dengan makna umum ,asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Asosiasi berarah/ <i>directed</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan/ <i>Dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Aggregasi/ <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian(whole-part)

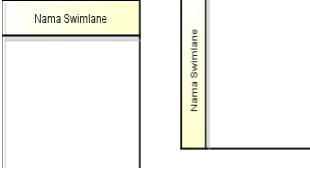
(Sumber: Rosa danShalahuddin, 2014)

2.12 Activity Diagram

Menurut Rosa dan Shalahuddin(2014) *activity diagram* Menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Perlu diperhatikan disini adalah bahwa *diagram* aktifitas menggambarkan aktifitas sistem bukan apa yang dilakukan aktor, jadi aktifitas yang dapat dilakukan oleh sistem.Simbol – simbol yang ada pada *activity diagram* dapat dilihat pada tabel berikut ini :

Tabel 2.4 Simbol *Activity Diagram*

Simbol	Keterangan
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

(Sumber : Rosa dan Shalahudhin, 2014)